

CLAIMS

What is claimed is:

1. A method of dynamically discovering and configuring a new network device comprising:
registering the new network device with a lookup service;
periodically reading the lookup service from an administration terminal;
responsive to the administration terminal detecting the new network device, notifying a
human operator of the presence of the new network device through a graphical
user interface on the administration terminal;
responsive to the human operator selecting an option available on the graphical user
interface, issuing a series of one or more generic Application Program Interface
(API) calls representative of the option to the new network device wherein said
API calls cause execution of interface software preloaded on the new network
device and contain instructions specific to the new network device for
implementing the API calls; and
executing the interface software preloaded on the new device to perform device specific
equivalents to the series of one or more generic API calls.
2. The method of claim 1, wherein said interface stored on the new network device
comprises a Java language program.

1 3. The method of claim 1, wherein said options available on the graphical user interface
2 comprise: create disk; create file system; delete disk; delete file system; and share file
3 functions.

1 4. The method of claim 1, wherein said device is a network attached storage device.

1 5. The method of claim 1, wherein said lookup service is the Jini lookup service.

1 6. A method of dynamically configuring a new network device comprising:
2 preloading the new network device with interface software wherein said interface
3 provides instructions specific to the new network device for executing a set of
4 generic Application Program Interface (API) calls;
5 registering the new device with a lookup service; and
6 responsive to receiving API calls, executing the interface software to perform device
7 specific equivalents to the generic API calls.

8 7. The method of claim 6, wherein said interface stored on the new network device
9 comprises a Java language program.

1 8. The method of claim 6, wherein said options available on the graphical user interface
2 comprise: create disk; create file system; delete disk; delete file system; and share file
3 functions.

1 9. The method of claim 6, wherein said device is a network attached storage device.

1 10. The method of claim 6, wherein said lookup service is the Jini lookup service.

1 11. A method of dynamically discovering and configuring a new network device comprising:
2 periodically reading a lookup service to determine whether a new device has been added
3 to the service;
4 responsive to finding a new device added to the service, reading interface data for the new
5 device from the lookup service and notifying a human operator of the presence of
6 the new network device through a graphical user interface on an administration
7 terminal; and
8 responsive to the human operator selecting options available on the graphical user
9 interface, sending a series of one or more generic Application Program Interface
10 (API) calls representative of the option to the new network device.

1 12. The method of claim 11, wherein said interface stored on the new network device
2 comprises a Java language program.

1 13. The method of claim 11, wherein said options available on the graphical user interface
2 comprise: create disk; create file system; delete disk; delete file system; and share file
3 functions.

1 14. The method of claim 11, wherein said device is a network attached storage device.

1 15. The method of claim 11, wherein said lookup service is the Jini lookup service.

1 16. A Storage Area Network (SAN) system comprising:
2 a network attached storage (NAS) device coupled with a network and preloaded with
3 interface software providing instructions specific to the NAS device for executing
4 a set of generic Application Program Interface (API) calls; and
5 an administration terminal coupled with said NAS device via the network and executing
6 software generating a graphical user interface and sending to the NAS device a
7 series of generic Application Program Interface (API) calls representative of
8 options selected from the graphical user interface wherein said API calls are
9 readable by the interface software of the NAS device.

10 17. The system of claim 16, wherein said interface stored on the NAS device comprises a
11 Java language program.

1 18. The system of claim 16, wherein said options available on the graphical user interface
2 comprise: create disk; create file system; delete disk; delete file system; and share file
3 functions.

1 19. A network attached storage (NAS) device coupled with a network and preloaded with
2 interface software providing instructions specific to the NAS device for executing a set of
3 generic Application Program Interface (API) calls.
4

5 20. The NAS device of claim 19, wherein said interface stored on the NAS device comprises
6 a Java language program.

1 21. The NAS device of claim 19, wherein said API calls represent calls for executing
2 commands including: create disk; create file system; delete disk; delete file system; and
3 share file functions.

1 22. A terminal coupled with a network executing software generating a graphical user
2 interface and sending to a NAS device a series of generic Application Program Interface
3 (API) calls representative of options selected from the graphical user interface wherein
4 said API calls are readable by an interface software the NAS

5 23. The terminal of claim 22, wherein said API calls represent calls for executing commands
6 including: create disk; create file system; delete disk; delete file system; and share file
7 functions.

1 24. A machine-readable medium having stored thereon data representing sequences of
2 instructions, the sequences of instructions which, when executed by a processor, cause the
3 processor to dynamically discover and configure a new network device by:
4 registering the new network device with a lookup service;
5 periodically reading the lookup service from an administration terminal;
6 responsive to the administration terminal detecting the new network device, notifying a
7 human operator of the presence of the new network device through a graphical
8 user interface on the administration terminal;
9 responsive to the human operator selecting options available on the graphical user
10 interface, sending a series of generic Application Program Interface (API) calls
11 representative of the option to the new network device wherein said API calls are

12 readable by interface software preloaded on the new network device and
13 containing instructions specific to the new network device for executing the API
14 calls; and
15 executing the interface software preloaded on the new device to perform device specific
16 equivalents to the generic API calls.

1 25. The machine-readable medium of claim 24, wherein said interface stored on the new
2 network device comprises a Java language program.

1 26. The machine-readable medium of claim 24, wherein said options available on the
2 graphical user interface comprise: create disk; create file system; delete disk; delete file
3 system; and share file functions.

1 27. The machine-readable medium of claim 24, wherein said device is a network attached
2 storage device.

1 28. The machine-readable medium of claim 24, wherein said lookup service is the Jini
2 lookup service.

1 29. A machine-readable medium having stored thereon data representing sequences of
2 instructions, the sequences of instructions which, when executed by a processor, cause the
3 processor to dynamically discover and configure a new network device by:

preloading the new network device with interface software wherein said interface provides instructions specific to the new network device for executing a set of generic Application Program Interface (API) calls; registering the new device with a lookup service; and responsive to receiving API calls, executing the interface software to perform device specific equivalents to the generic API calls.

30. The machine-readable medium of claim 29, wherein said interface stored on the new network device comprises a Java language program.

31. The machine-readable medium of claim 29, wherein said options available on the graphical user interface comprise: create disk; create file system; delete disk; delete file system; and share file functions.

32. The machine-readable medium of claim 29, wherein said device is a network attached storage device.

33. The machine-readable medium of claim 29, wherein said lookup service is the Jini lookup service.

34. A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instructions which, when executed by a processor, cause the processor to dynamically discover and configure a new network device by:

periodically reading a lookup service to determine whether a new device has been added to the service;
responsive to finding a new device added to the service, reading interface data for the new device from the lookup service and notifying a human operator of the presence of the new network device through a graphical user interface on the administration terminal; and
responsive to the human operator selecting options available on the graphical user interface, sending a series of generic Application Program Interface (API) calls representative of the option to the new network device.

35. The machine-readable medium of claim 34, wherein said interface stored on the new network device comprises a Java language program.

36. The machine-readable medium of claim 34, wherein said options available on the graphical user interface comprise: create disk; create file system; delete disk; delete file system; and share file functions.

37. The machine-readable medium of claim 34, wherein said device is a network attached storage device.

38. The machine-readable medium of claim 34, wherein said lookup service is the Jini lookup service.